WHAT IS CLAIMED IS:

1   1.   A method for generating a color value for a pixel from geometry data,
2   the method comprising:
3   selecting a first plurality of shading locations and a second plurality of depth
4   locations for the pixel, the second plurality being larger in number than the first plurality,
5   each of the second plurality of depth locations being associated with one of the shading
6   locations;
7   under control of a graphics processing subsystem, generating a plurality of
8   hybrid sampled data points equal in number to the second plurality of depth locations,
9   wherein the act of generating includes:
10   computing a shading value at each of the first plurality of shading
11   locations and a depth value at each of the second plurality of depth locations; and
12   storing one of the depth values and the associated shading value as one
13   of the hybrid sampled data points; and
14   computing an antialiased color value for the pixel using the hybrid sampled
15   data points.

1   2.   The method of claim 1 wherein the act of generating the plurality of
2   hybrid sampled data points includes:
3   performing at least two multisampling operations on the pixel,
4   wherein each multisampling operation uses a different one of the shading
5   locations and a different subset of the depth locations and generates a different subset of the
6   plurality of hybrid sampled data points.

1   3.   The method of claim 2 wherein the subset of the hybrid sampled data
2   points generated by each multisampling operation is stored in a corresponding one of a
3   plurality of target buffers.

1   4.   The method of claim 2 wherein the subset of the hybrid sampled data
2   points generated by each multisampling operation is accumulated in an accumulation buffer.

1   5.   The method of claim 1 wherein each of the depth locations is inside the
2   pixel.

1   6. The method of claim 1 wherein each of the shading locations is inside

2 the pixel.

1   7. The method of claim 1 wherein the geometry data includes a primitive,

2 the method further comprising, prior to storing one of the depth values and the associated

3 shading value, determining whether the primitive covers the depth location,

4   wherein the one of the depth value and the associated one of the shading value

5 are not stored in the event that the primitive does not cover the depth location.

1   8. The method of claim 1 wherein the act of selecting the first plurality of

2 shading locations and the second plurality of depth locations for the pixel includes:

3   segmenting a viewable area that includes the pixel into a number of

4 sub-pixels, each sub-pixel having a size smaller than a size of the pixel,

5   wherein each sub-pixel includes one of the shading locations and a subset of

6 the depth locations.

1   9. The method of claim 8 wherein associating each of the second plurality

2 of depth locations with one of the shading locations includes:

3   associating each of the depth locations of a sub-pixel with the shading location

4 of that sub-pixel.

1   10. The method of claim 8 wherein the pixel is divided into an integer

2 number of sub-pixels.

1   11. The method of claim 8 wherein the act of segmenting the viewable

2 area includes providing a multisampling rasterizer with a display resolution that is larger than

3 a true display resolution.

1   12. The method of claim 10 wherein the pixel is divided into four

2 sub-pixels arranged to form a quad.

1   13. The method of claim 1 wherein the act of selecting the first plurality of

2 shading locations and the second plurality of depth locations for the pixel includes:

3   defining a multisampling pattern for the pixel, the multisampling pattern

4 including one of the depth locations and at least two of the shading locations;

5   generating a plurality of iterations of the geometry data, wherein each iteration

6   has a different offset relative to a boundary of the pixel; and

7   applying the multisampling pattern to each of the iterations of the geometry

8   data.

1   14.   The method of claim 13 wherein each of the offsets corresponds to an

2   amount less than a pixel size.

1   15.   The method of claim 14 wherein one of the offsets is equal to zero.

1   16.   The method of claim 13 wherein the act of generating the plurality of

2   iterations includes, for each iteration, setting a value of a viewport offset parameter

3   corresponding to the offset of the iteration.

1   17.   The method of claim 13 wherein generating the plurality of hybrid

2   sampled data points includes:

3   storing the depth values and the associated shading value obtained from each

4   iteration in a respective one of a plurality of buffers.

1   18.   The method of claim 1 wherein the act of selecting the first plurality of

2   shading locations and the second plurality of depth locations for the pixel includes:

3   defining a multisampling pattern for the pixel, the multisampling pattern

4   including one of the depth locations and at least two of the shading locations;

5   defining a plurality of non-overlapping regions in an image coordinate space,

6   each region including a virtual pixel corresponding to the pixel;

7   relocating the geometry data to a position within each of the regions, wherein

8   the position of the relocated geometry data relative to a boundary of the region is shifted by

9   an amount less than a pixel size; and

10   applying the multisampling pattern to each of the virtual pixels.

1   19.   The method of claim 18 wherein one of the regions corresponds to a

2   viewable area of the image coordinate space.

1   20.   The method of claim 18 wherein for one of the regions, the amount

2   less than a pixel size is zero.

1   21.   The method of claim 18 wherein relocating the geometry data includes:

2   setting a value of a window offset parameter such that the geometry data is

3   placed within one of the regions; and

4   setting a value of a viewport offset parameter corresponding to the shift by an

5   amount less than a pixel size.

1   22.   The method of claim 18 wherein the act of relocating the geometry

2   data is performed by the graphics processing subsystem.

1   23.   The method of claim 1 wherein the act of computing the color value

2   for the pixel includes:

3   defining a texture map including a second plurality of texels corresponding to

4   the hybrid sampled data points for the pixel;

5   fetching the second plurality of texels; and

6   computing a weighted average of the fetched texels, thereby determining the

7   color value for the pixel.

1   24.   The method of claim 1 wherein the act of computing the color value

2   for the pixel includes:

3   defining a plurality of texture maps, each texture map including a plurality of

4   texels corresponding to a subset of the hybrid sampled data points for the pixel;

5   for each of the plurality of texture maps:

6   fetching the plurality of texels from the texture map; and

7   blending the fetched texel values to generate an intermediate value;

8   and

9   computing a weighted average of the intermediate value generated for each of

10   the texture maps, thereby determining the color value for the pixel.

1   25.   The method of claim 1 wherein the act of computing the color value

2   for the pixel is performed during a scanout operation that provides downfiltered color data to

3   a display device.

1   26.   The method of claim 1 wherein the acts of generating the plurality of

2   hybrid sampled data points and computing the color value for the pixel are performed in a

3   single rendering pass.

1   27.   The method of claim 1 wherein the number of shading locations and
2   the number of depth locations are determined based on one or more configurable parameters.

1   28.   A system for generating a color value for a pixel from geometry data,
2   the system comprising:
3   a multisampling rasterizer configured to receive the geometry data and
4   perform a multisampling operation on the pixel, the multisampling operation generating a
5   plurality of depth values at a plurality of depth locations for the pixel and one shading value,
6   the shading value being associated with each of the plurality of depth locations;
7   control logic configured to use the multisampling rasterizer to perform a
8   plurality of multisampling operations on the pixel; and
9   a downfiltering unit configured to combine the shading values generated
10 during the plurality of multisampling operations, thereby generating a color value for the
11 pixel.

1   29.   The system of claim 28 wherein the control logic is further configured
2   to select different depth locations for each of the plurality of multisampling operations.

1   30.   The system of claim 28 wherein the control logic is further configured
2   to change a screen location of the geometry data such that the multisampling rasterizer uses
3   different depth locations for each of the plurality of multisampling operations.

1   31.   The system of claim 28 wherein the control logic is further configured
2   to use the multisampling rasterizer to perform a multisampling operation on the geometry
3   data for each of a plurality of sub-pixels at different locations within the pixel.

1   32.   The system of claim 31 wherein the multisampling rasterizer is
2   instructed to use a display resolution larger than a true display resolution.

1   33.   The system of claim 28 wherein the control logic is further configured
2   to store the geometry data and to supply the geometry data to the multisampling rasterizer
3   multiple times in succession.

1   34.   The system of claim 28 wherein the control logic is further configured
2   to relocate the geometry data in each of a plurality of non-overlapping regions and to instruct

3   the multisampling rasterizer to perform a multisampling operation on a virtual pixel in each
4   region.

1   35.   The system of claim 34 further comprising a buffer having a plurality
2   of non-overlapping regions, wherein multisampled pixel data from each of the
3   non-overlapping region is stored in a respective one of the non-overlapping regions.

1   36.   The system of claim 34 further comprising a plurality of buffers,
2   wherein multisampled pixel data from each of the non-overlapping regions is stored in a
3   respective one of the plurality of buffers.

1   37.   The system of claim 28 further comprising:
2   a frame buffer for storing the shading value at each depth location,
3   wherein the downfiltering unit is further configured to read the shading values
4   from the frame buffer.

1   38.   The system of claim 37 wherein the downfiltering unit includes:
2   a texture processing unit configured to fetch at least one of the shading values
3   from the frame buffer as a texel and to generate an intermediate value from the texel; and
4   a shader configured to blend the intermediate values, thereby generating the
5   color value for the pixel.

1   39.   The system of claim 38 wherein:
2   the texture processing unit is further configured to fetch all of the shading
3   values for the pixel from the frame buffer and to provide each fetched shading value as an
4   intermediate value.

1   40.   The system of claim 38 wherein:
2   the texture processing unit is further configured to fetch a plurality of subsets
3   of the shading values for the pixel from the frame buffer and to blend each subset of the
4   shading values, thereby generating a plurality of intermediate values.

1   41.   An apparatus for generating a color value for a pixel from geometry
2   data, the apparatus comprising:
3   a graphics processor including:

4   a multisampling rasterizer configured to receive the geometry data and

5   perform a multisampling operation on the pixel, the multisampling operation

6   generating a plurality of depth values at a plurality of depth locations for the pixel and

7   one shading value, the shading value being associated with each of the plurality of

8   depth locations;

9   control logic configured to use the multisampling rasterizer to perform

10   a plurality of multisampling operations on the pixel; and

11   a downfiltering unit configured to combine the shading values

12   generated during the plurality of multisampling operations, thereby generating a color

13   value for the pixel;

14   a frame buffer configured to store the shading values generated during the

15   plurality of multisampling operations; and

16   a downfiltering unit configured to combine the shading values stored in the

17   frame buffer, thereby generating a color value for the pixel.

1   42.   The apparatus of claim 41 further comprising:

2   a graphics driver module configured to communicate with the graphics

3   processor and to configure a parameter for the plurality of multisampling operations.

1   43.   The apparatus of claim 42 wherein the parameter determines a number

2   of multisampling operations to be performed.

1   44.   The apparatus of claim 42 wherein the parameter determines a number

2   of depth locations to be used during each of the multisampling operation.

1   45.   The apparatus of claim 42 wherein the graphics driver module includes

2   an application program interface for configuring the parameter.

1   46.   The apparatus of claim 42 wherein the graphics driver module includes

2   a user interface for configuring the parameter.

1   47.   The apparatus of claim 42 wherein the graphics driver module is

2   further configured to detect a property of an application program and to configure the

3   parameter based at least in part on the detected property.